59

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/964,763	09/27/2001	Arch D. Robison	042390.P11908	2972

8791 7590 01/07/2005

BLAKELY SOKOLOFF TAYLOR & ZAFMAN
12400 WILSHIRE BOULEVARD
SEVENTH FLOOR
LOS ANGELES, CA 90025-1030

EXAMINER	
YIGDALL, MICHAEL J	
ART UNIT	**PAPER NUMBER**
2122	

DATE MAILED: 01/07/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	**Application No.** 09/964,763	**Applicant(s)** ROBISON, ARCH D.	
	Examiner Michael J. Yigdall	**Art Unit** 2122	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *18 October 2004*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-7,13-21 and 27-36* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-7,13-21 and 27-36* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date _____.

4) ☐ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ .

5) ☐ Notice of Informal Patent Application (PTO-152)

6) ☐ Other: _____.

DETAILED ACTION

1. A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after final rejection. Since this application is eligible for continued examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114. Applicant's submission filed on October 18, 2004 has been entered. Claims 1-7, 13-21 and 27-36 are now pending.

Response to Arguments

2. Applicant's arguments have been fully considered but they are not persuasive.

3. As set forth in the advisory action mailed on September 23, 2004, and in response to Applicant's arguments against the references individually (Applicant's remarks, pages 13 and 14), one cannot show nonobviousness by attacking references individually where the rejections are based on combinations of references. See *In re Keller*, 642 F.2d 413, 208 USPQ 871 (CCPA 1981) and *In re Merck & Co.,* 800 F.2d 1091, 231 USPQ 375 (Fed. Cir. 1986).

4. Applicant contends that Archambault does not disclose determining whether at least one pointer is aliased with at least one restricted pointer when the at least one restricted pointer is out-of-scope relative to the at least one pointer. Specifically, Applicant contends that Archambault does not disclose alias analysis of restricted pointers, and that Archambault does not disclose determining an alias with a restricted pointer that is out-of-scope relative to another pointer (Applicant's remarks, page 13, second paragraph).

However, Archambault discloses both intraprocedural alias analysis (see, for example, column 5, lines 4-17) and interprocedural alias analysis (see, for example, column 7, lines 27-33). As Applicant acknowledges, during intraprocedural analysis, a pointer alias graph is built for each function. Each function or procedure has a local scope, which is the scope for the pointers within that function or procedure (see, for example, column 5, lines 52-56). During interprocedural analysis, Archambault then determines alias sets based on a universal pointer alias graph built for the whole program (see, for example, column 7, lines 34-45). Therefore, Archambault finds aliases of pointers from more than one procedure, which is to say aliases of pointers from more than one scope. In other words, Archambault discloses determining an alias with a pointer that is out-of-scope relative to another pointer.

Blainey too discloses alias analysis, for optimization in a compiler (see, for example, column 3, lines 26-29). Blainey further discloses that precise alias information can be determined based on language rules, language features and assertions made by the programmer (see, for example, column 2, lines 40-46). It should be noted that Archambault similarly discloses that the size of alias sets can be reduced with alias assertions made by the programmer (see, for example, column 2, lines 43-45), and that precise alias sets can be used to improve optimization (see, for example, column 3, lines 12-18).

Robison discloses an example of the language features suggested by Blainey, specifically restricted pointers and the use of the "restrict" keyword as an assertion made by the programmer in pointer declarations (see, for example, the "Restrict Qualifies Pointers" section). Robison further discloses that restricted pointers address problems associated with aliases and improve the performance of programs written in the C and C++ languages (see, for example, the

"FORTRAN Envy" and "Conclusion" sections). It should be noted that Robison also discloses that the meaning and use of the "restrict" keyword is associated with the scope of the pointers (see, for example, the "Restrict Qualifies Pointers" section).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to supplement the alias analysis system of Archambault with language rules, language features and programmer assertions, so as to precisely determine aliases and reduce the size of alias sets, as taught by Blainey and as suggested by Archambault, including the language feature of restricted pointers disclosed by Robison, for the purpose of improving optimization and improving the performance of programs.

As presented above, Archambault discloses determining whether at least one pointer is aliased with another pointer when the other pointer is out-of-scope relative to the at least one pointer. Therefore, in combination, Archambault, Blainey and Robison disclose determining whether at least one pointer is aliased with at least one restricted pointer when the at least one restricted pointer is out-of-scope relative to the at least one pointer.

5. Applicant contends that Blainey does not disclose or suggest determining whether at least one pointer is aliased with at least one restricted pointer when the at least one restricted pointer is out-of-scope relative to the at least one pointer (Applicant's remarks, page 14, second paragraph). However, Archambault, Blainey and Robison disclose this feature in combination, as presented above.

6. Applicant contends that Robison does not discuss a method for optimizing C and C++ code through alias analysis that includes restricted pointers, and that Robison consequently does

not disclose or suggest determining whether at least one pointer is aliased with at least one restricted pointer when the at least one restricted pointer is out-of-scope relative to the at least one pointer (Applicant's remarks, page 14, third paragraph). However, again, Archambault, Blainey and Robison disclose this feature in combination, as presented above. Moreover, Robison discloses that restricted pointers address aliasing problems and improve the performance of programs, also as presented above.

Claim Rejections - 35 USC § 103

7. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

8. Claims 1-7, 13-21 and 27-36 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Pat. No. 6,173,444 to Archambault (art of record, "Archambault" herein) in view of U.S. Pat. No. 6,045,585 to Blainey (art of record, "Blainey" herein) in view of "Restricted Pointers are Coming" by Robison (art of record, "Robison" herein).

With respect to claim 1 (currently amended), Archambault discloses a method comprising:

(a) receiving a code segment having a plurality of instructions, the code segment having an outer scope and a number of inner scopes, wherein the plurality of instructions comprise a number of pointers (see, for example, column 5, lines 4-23, which shows receiving program code having a number of functions, i.e. a number of inner scopes within the global or outer scope of the program, and a plurality of instructions comprising a number of pointers).

Although Archambault discloses pointer variables in the C programming language (see, for example, column 4, lines 13-15), Archambault does not expressly disclose the limitation wherein at least one of the number of pointers is a restricted pointer.

However, Archambault discloses that the size of alias sets can be reduced with alias assertions made by the programmer (see, for example, column 2, lines 43-45), and that precise alias sets can be used to improve optimization (see, for example, column 3, lines 12-18).

Moreover, Blainey discloses that precise alias information can be determined based on language rules, language features and assertions made by the programmer (see, for example, column 2, lines 40-46). Blainey teaches a system for program optimization in a compiler (see, for example, column 3, lines 26-29).

Robison discloses an example of the language features suggested by Blainey, specifically restricted pointers and the use of the "restrict" keyword as an assertion made by the programmer in pointer declarations (see, for example, the "Restrict Qualifies Pointers" section). Robison further discloses that restricted pointers address problems associated with aliases and improve the performance of programs written in the C and C++ languages (see, for example, the "FORTRAN Envy" and "Conclusion" sections).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to supplement the alias analysis system of Archambault with language rules, language features and programmer assertions, so as to precisely determine aliases and reduce the size of alias sets, as taught by Blainey and as suggested by Archambault, including the language feature of restricted pointers disclosed by Robison, for the purpose of improving optimization and improving the performance of programs.

Therefore, Archambault in view of Blainey in view of Robison discloses the limitation wherein at least one of the number of pointers is a restricted pointer.

Archambault also discloses:

(b) determining whether at least one pointer of the number of pointers is aliased with the at least one restricted pointer when the at least one restricted pointer is out-of-scope relative to the at least one pointer (see, for example, column 7, lines 27-45, which shows interprocedural alias analysis, i.e. determining whether a pointer is aliased with a pointer in another procedure, and column 5, lines 52-56, which shows that each function or procedure has a local scope, i.e. a pointer in one procedure is out-of-scope relative to a pointer in another procedure).

With respect to claim 2 (original), Archambault also discloses determining a base pointer for each pointer of the number of pointers (see, for example, column 5, lines 4-17, which shows determining a base pointer for each pointer definition comprising the right-hand side of its associated variable assignment operation).

With respect to claim 3 (original), Archambault also discloses the limitation wherein the determining a base pointer for each pointer of the number of pointers comprises grouping pointers together upon determining that the pointers are copied to a pointer that is not a restricted pointer (see, for example, column 6, lines 16-46, which shows finding the union of alias sets and propagating transitive relationships to group pointers together, when appropriate, such as when pointers are copied to a non-restricted pointer).

With respect to claim 4 (original), Archambault also discloses the limitation wherein there is no grouping of pointers when the pointers have distinct base pointers (see, for example, column 5, lines 31-41, which shows adding new nodes to the pointer graph, i.e. not grouping the pointers into an alias set, when the base pointers are distinct and thus not already represented in the graph).

With respect to claim 5 (original), Archambault also discloses, for each instruction of the plurality of instructions that accesses a pointer, determining which at least one restricted pointer is within the scope of the pointer when the pointer is accessed (see, for example, column 5, lines 52-56, which shows determining the pointer variables accessed in the local scope).

With respect to claim 6 (currently amended), Archambault also discloses the limitation wherein the determining whether at least one pointer of the number of pointers is aliased with the at least one restricted pointer is based on the base pointer for each of the number of pointers (see, for example, column 6, lines 61-67, which shows that alias sets are determined based on the L-value, i.e. the memory address or base pointer).

With respect to claim 7 (currently amended), Archambault also discloses the limitation wherein the determining whether at least one pointer of the number of pointers is aliased with the at least one restricted pointer is based on, for each instruction of the plurality of instructions that accesses the pointer, which at least one restricted pointer is within the scope of the pointer, when the pointer is accessed (see, for example, column 5, lines 52-56, which shows determining the alias sets for all pointer variables accessed in the local scope).

With respect to claim 13 (currently amended), the system recited in the claim corresponds to the method recited in claim 1 (see Archambault, Blainey and Robison as applied to claim 1 above). Note that Archambault also discloses a compiler coupled to a memory unit (see, for example, column 3, lines 12-18).

With respect to claim 14 (original), the limitations recited in the claim are analogous to the limitations recited in claim 2 (see Archambault, Blainey and Robison as applied to claim 2 above).

With respect to claim 15 (original), the limitations recited in the claim are analogous to the limitations recited in claim 5 (see Archambault, Blainey and Robison as applied to claim 5 above).

With respect to claim 16 (currently amended), the limitations recited in the claim are analogous to the limitations recited in claim 7 (see Archambault, Blainey and Robison as applied to claim 7 above).

With respect to claim 17 (currently amended), the product recited in the claim corresponds to the method recited in claim 1 (see Archambault, Blainey and Robison as applied to claim 1 above). Note that Archambault also discloses a machine-readable medium that provides instructions to be executed by a machine (see, for example, column 3, lines 36-40).

With respect to claim 18 (original), the limitations recited in the claim are analogous to the limitations recited in claim 2 (see Archambault, Blainey and Robison as applied to claim 2 above).

With respect to claim 19 (original), the limitations recited in the claim are analogous to the limitations recited in claim 5 (see Archambault, Blainey and Robison as applied to claim 5 above).

With respect to claim 20 (currently amended), the limitations recited in the claim are analogous to the limitations recited in claim 6 (see Archambault, Blainey and Robison as applied to claim 6 above).

With respect to claim 21 (currently amended), the limitations recited in the claim are analogous to the limitations recited in claim 7 (see Archambault, Blainey and Robison as applied to claim 7 above).

With respect to claim 27 (new), Archambault also discloses the limitation wherein each of the at least two pointers are aliases for the same memory location if the at least two pointers have the same base pointer (see, for example, column 2, lines 16-33, which shows that pointers are aliases for the same memory location if the pointers have the same base pointer).

With respect to claim 28 (new), Archambault also discloses the limitation wherein if the at least two pointers do not have the same base pointers:

(a) determining whether each base pointer of the at least two pointers is a restricted pointer (see, for example, column 7, lines 13-19, which shows determining types of pointers, such as restricted pointers).

Although Archambault discloses determining when pointers are in scope when the pointers are used, i.e. indirectly read or written (see, for example, column 5, lines 52-56), Archambault does not expressly disclose:

(b) if each base pointer is a restricted pointer, determining whether any of the base pointers is in scope when the other base pointers are indirectly read or written.

However, Robison discloses that the meaning and use of restricted pointers is associated with the scope of the pointers (see, for example, the "Restrict Qualifies Pointers" section).

As presented above, it would have been obvious to one of ordinary skill in the art at the time the invention was made to supplement the alias analysis system of Archambault with language rules, language features and programmer assertions, so as to precisely determine aliases and reduce the size of alias sets, as taught by Blainey and as suggested by Archambault, including the language feature of restricted pointers disclosed by Robison, for the purpose of improving optimization and improving the performance of programs.

Therefore, Archambault in view of Blainey in view of Robison discloses the limitation wherein if each base pointer is a restricted pointer, determining whether any of the base pointers is in scope when the other base pointers are indirectly read or written.

With respect to claim 29 (new), Archambault also discloses the limitation wherein each of the at least two pointers are not aliases for the same memory location if each of the base pointers of the at least two pointers are in scope when the other base pointers are indirectly read or written (see, for example, column 2, lines 16-33, which shows that pointers are not aliases for the same memory location if the base pointers are in scope when other base pointers are dereferenced, i.e. indirectly read or written).

With respect to claim 30 (new), Archambault also discloses the limitation wherein if any base pointer is not a restricted pointer, determining whether each base pointer of the at least two pointers is a restricted pointer or is a parameter pointer (see, for example, column 7, lines 16-33, which shows determining types of pointers, such as restricted pointers and argument or parameter pointers).

With respect to claim 31 (new), Archambault also discloses the limitation wherein each of the at least two pointers are aliases for the same memory location if at least one base pointer is not a restricted pointer or is not a parameter pointer (see, for example, column 7, lines 16-33, which shows determining aliases according to the types of pointers, such as restricted pointers and argument or parameter pointers).

With respect to claim 32 (new), the limitations recited in the claim are analogous to the limitations recited in claim 27 (see Archambault, Blainey and Robison as applied to claim 27 above).

With respect to claim 33 (new), the limitations recited in the claim are analogous to the limitations recited in claim 28 (see Archambault, Blainey and Robison as applied to claim 28 above).

With respect to claim 34 (new), the limitations recited in the claim are analogous to the limitations recited in claim 29 (see Archambault, Blainey and Robison as applied to claim 29 above).

With respect to claim 35 (new), the limitations recited in the claim are analogous to the limitations recited in claim 30 (see Archambault, Blainey and Robison as applied to claim 30 above).

With respect to claim 36 (new), the limitations recited in the claim are analogous to the limitations recited in claim 31 (see Archambault, Blainey and Robison as applied to claim 31 above).

Conclusion

9. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Michael J. Yigdall whose telephone number is (571) 272-3707. The examiner can normally be reached on Monday through Friday from 7:30am to 4:00pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Q. Dam can be reached on (571) 272-3695. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

MY

Michael J. Yigdall
Examiner
Art Unit 2122

mjy

TUAN DAM
SUPERVISORY PATENT EXAMINER